Surv. Math. Ind. (1997) 6: 289–310

Surveys on Mathematics for Industry

A survey of grid generation techniques and systems with emphasis on recent developments

Joe F. Thompson, Mississippi State and **Bernd Hamann,** Davis

Summary. This survey reviews some of the most important grid generation techniques with emphasis on current trends. We provide a list of grid generation systems in use today and briefly discuss their capabilities. The most recent literature on the subject matter is discussed.

AMS Subject Classification: 65D17, 65M06, 65N06, 65N50, 65N55, 68U05.

Keywords: Approximation; Computational fluid dynamics; Computer-aided design; Grid generation; Grid generation systems; Structured grids; Unstructured grids; Adaptive grids.

1 Introduction

Grid (mesh) generation is an enabling computational technology in the construction of computational simulations of physical field phenomena for scientific investigation and for engineering design and analysis. The grid is the discrete structure on which the finite representation of the governing partial differential equations is made. This discrete representation may be finite element, finite volume, or finite differences—in any case there is an underlying grid discretizing the field into a collection of finite cells defined by associated grid points. Grids are not fundamentally peculiar to any of these forms of representation, although some associations have historically been made.

The grid generation process generally proceeds from a CAD (computer-aided design) consisting of the surfaces defining the geometry at hand, or from a discretization of these surfaces, to first create a grid on the boundary surfaces and then to fill the surrounding, or enclosed, volume with a grid. Ideally, this process involves little user interaction—with what user interaction there is being through a GUI (graphical user interface). And for a grid generation system to be really useful in the design mode, it must be very easy to introduce geometrical changes in the boundaries and modify the grid accordingly. Finally, the grid should be coupled dynamically with the solution on the grid: able to automatically recognize developing solution gradients or other regions in need of higher resolution.

Grid generation, unfortunately, remains a pacing item of CFD (computational fluid dynamics) and, in more general terms, CFS (computational field simulation)—not so much because of the need for new mathematical developments but because of the logical difficulties in making the generation process automatic for completely general geometrical configurations. To document that pacing item cliche, here are a few recent general observations from industry (with emphasis added).

From Ray Cosner [25] at McDonnell Douglas:

“Surface modeling and grid generation technology have long been recognized as a *critical issue* in practical applications of CFD analyses.

Surface modeling tools have gained great sophistication in the last ten years. However, the interface to the subsequent CFD analysis codes often is *cumbersome* and *restrictive*.

Without improvements in the process, we can expect the surface modeling and grid generation phase of the process to become a far worse *bottleneck* in the next few years.

The need for specialized skills is a potential *choke point* in the process.

This work prior to running the flow solver code consumed about 80% of the total manhours! Clearly, in reducing the manhours and thus the direct cost of CFD analysis we should focus on the tasks of handling the geometry and building the grid.”

Stuart Connell et al. [24] at General Electric comment:

“When computing the flow around complex three-dimensional configurations, the generation of the mesh is the most *time-consuming* part of any calculation.”

One requirement that comes through loud and clear is that it must be “very simple to change the geometry” in the design process. Raj [96] notes that:

“The most challenging situation arises when the configuration geometry undergoes changes and multiple analyses have to be performed for each variation. However, that is precisely what IPPD (Integrated Product and Process Development) design environment requires of CFD!”

Correcting input geometry defects in the CAD representation is a major bottleneck in the process. The grid generation system must be able to remove unwanted detail from the incoming CAD data, and to fix defects such as overlaps and gaps in the CAD model. Yet the quality of the representation of the true surface geometry can be critical to detailed CFD analysis. There must therefore be some CAD-type capability within the grid generation system, but with a direct link to the incoming CAD model. Without influence on the CAD system vendors—and that has been slow in coming—all this falls by default to the grid generation system: strong motivation for interaction with those vendors. In this regard, it should be appreciated that surface modeling and grid generation extend beyond CFD to all field problems throughout the design process, and in fact in today’s multidisciplinary design optimization, all are linked from the beginning.

There has been a series of international conferences on grid generation: Landshut (Germany) 1985, Miami (Florida, U.S.A.) 1988, Barcelona (Spain) 1990, Swansea (Wales, U.K.) 1993, and Starkville (Mississippi, U.S.A.) 1996, all of which have published proceedings (Numerical grid generation in computational field simulation, edited by B. K. Soni, J. Hauser, P. R. Eiseman, and J. F. Thompson, 1996; Numerical grid generation in computational fluid dynamics and related fields, edited by N. P. Weatherill, P. R. Eiseman, J. Hauser, and J. F. Thompson, 1994; Numerical grid generation in computational fluid dynamics and related fields, edited by A. S. Arcilla, J. Hauser, P. R. Eiseman, and J. F. Thompson, 1991; and Numerical grid generation in computation fluid dynamics '88, edited by S. Sengupta, J. Hauser, P. R. Eiseman, J. F. Thompson, 1988). There also have been three NASA conferences on the subject (Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop, 1995 and Proceedings of the Software Systems for Surface Modeling and Grid Generation Workshop, edited by R. E. Smith, 1992). Several surveys have been given [36, 128, 129]. The original textbook in the area appeared in 1985 [135], and other books have since come out [19, 46, 75]. A tutorial was given by Thompson and Weatherill [130], and chapters on the topic are included in several books.

This present paper gives a general overview and notes several large grid generation systems that have emerged.

2 General approaches

In this section, we discuss the main techniques used in numerical grid generation and provide a classification of the resulting grid types (see Fig. 1). Essentially, one can distinguish between two grid types:

- Structured grids, classically generated by a combination of transfinite interpolation (TFI) and solving elliptic (or hyperbolic) systems of partial differential equations (PDES).
- Unstructured grids, typically generated by a triangulation/tesselation algorithm (e.g., Delaunay triangulation and Voronoi diagram) or an advancing front approach.

In the grid generation community, the term *structured grid* is associated with a grid whose constituting elements (or cells) are topologically equivalent to a square (2D) or a cube (3D). In grid generation, the terms quadrilateral and hexahedron usually refer to a continuously deformed square or cube. When discretizing a single deformed square or cube by a structured grid, one usually chooses a curvilinear grid, i.e., a grid whose edges are all straight line segments.

The connectivity among nodes in a structured grid is completely defined by the nodes' indices, e.g., a node $\mathbf{x}_{i,j}$ in a 2D structured grid is connected with the nodes $\mathbf{x}_{i-1,j}$, $\mathbf{x}_{i+1,j}$, $\mathbf{x}_{i,j-1}$, and $\mathbf{x}_{i,j+1}$. This implied connectivity helps in designing highly efficient solution algorithms for the PDES describing some physical phenomenon.

The term *unstructured grid* refers to any kind of grid that is not a structured grid. The only unstructured grids of practical importance consist of triangles and/or quadrilaterals (2D) and tetrahedra, pentahedra (prisms), and/or hexahedra

(3D). Thus, quadrilaterals and hexahedra can constitute an unstructured grid, but the number of edges sharing a node is not restricted.

Combinations of structured and unstructured grid types are used as well. The most common grid types and combinations of various grid types are:

- Structured grids, consisting of quadrilateral and hexahedral elements whose node connectivity is implicitly defined by the nodes' indices.
- Block-structured grids, consisting of multiple structured grids, each one associated with one of many blocks (connectivity among blocks not necessarily structured).
- Unstructured grids, consisting of quadrilateral, triangular, hexahedral, pentahedral, tetrahedral, and other types of polygonal/polyhedral elements; node connectivity explicitly defined for each node.
- Hybrid grids, consisting of structured and unstructured grid regions.
- Chimera (overset) grids, consisting of multiple structured grids with partially overlapping grid elements; overlap regions typically "resolved" using appropriate interpolation schemes.
- Hierarchical grids, generated by quadtree- and octree-like subdivision schemes (also referred to as "embedded" grid or "semi-structured" grids).

These grid types are illustrated for 2D in Fig. 1.

More information about these and other existing grid types and generation techniques/systems is provided by Castillo [19] and others [46, 51, 75, 135, 129]. Thompson [128] provides a large set of recent references in the area, and the tutorial given by Thompson and Weatherill [130] provides a general introduction to grid generation.

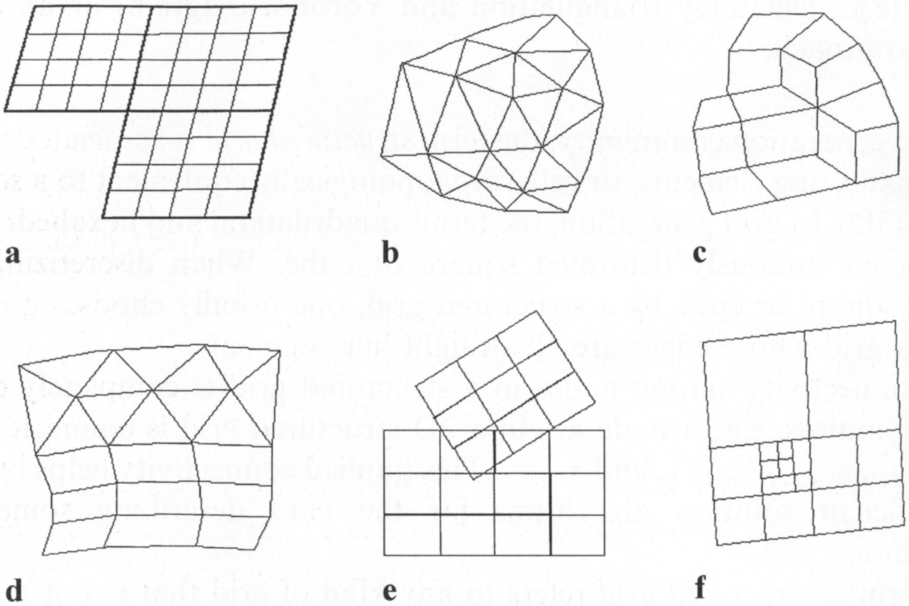

Fig. 1. **a** Multi-block structured, **b** unstructured-triangular, **c** unstructured-quadrilateral, **d** hybrid, **e** Chimera, and **f** hierarchical grid

2.1. *Block-structured grids*

Block-structured grids opened the door to real-world CFD in the late 80s, and most real applications are still based on these grids. Although the grid is topologically rectangular within each block, the blocks fit together in an unstructured manner, i.e, more or less than eight blocks might share a common corner point. Complete continuity across block interfaces in the field is accomplished by treating the interface in the manner of a branch cut, correspondence between points outside one block with points inside the adjacent block.

Structured grid generation using TFI and elliptic PDEs

One of the classical approaches for generating block-structured 2D (3D) grids for a finite space surrounding a geometry is based on TFI [35, 47] and solving elliptic systems of PDEs [135]. Once the block configuration around a geometry is established, an initial grid is generated inside each block by performing bilinear (trilinear) TFI of each block's boundary edges (faces). The TFI algorithm is performed in a discrete manner for a finite set of points on the boundary edges (faces) of each block. The points in the boundary edges (faces) are spaced according to a specified distribution function (e.g., uniform in parameter space, uniform with respect to arc length, uniform with respect to integrated absolute curvature, etc.).

The now-standard procedure in block-structured systems is to first generate surface grids on block faces—both boundary and in-field block interfaces—from point distributions placed on the face edges by these distribution functions. The volume grids are generated within the blocks. In both this surface and volume grid generation, the first step is normally TFI to be followed by elliptic generation (solution of an elliptic system of PDEs for the grid, done in topologically cartesian computational space) with control functions interpolated into the field in accordance with boundary point distribution and surface curvature. The elliptic system provides smoothing, with control of spacing and orientation, and need not be iterated to convergence.

Orthogonality and control of off-boundary spacing at boundaries are desirable for the treatment of boundary layers and in the implementation of turbulence models, and the standard approach used to achieve these features in the grid has been the iterative adjustment of the control functions in the elliptic generation system.

Technical details of block-structured grid generation, both by TFI and by elliptic systems, are given by Thompson [133, 134]. Some recent innovations are given by Spekreijse [118]. Another approach to boundary orthogonality is through higher-order elliptic systems [116].

Elliptic generation systems operate throughout the entirety of a region, while hyperbolic systems move outward from boundaries, generating an orthogonal grid, but without matching an outer boundary. Good sources for details are Steger [121] and Chan and Steger [21].

Overset grids

The Chimera (overset) approach has great versatility and is especially attractive with bodies in relative motion. Here structured grids are generated, generally by a hyperbolic system, outward a relatively small distance from each boundary surface and are overset on a background grid and one another, with communication among these component grids via interpolation. The method has, however, never attracted as many users as might have been expected. Concerns have been continually raised about the accuracy of the interpolation necessary to transfer data between component grids, and particularly the lack of conservation that is attendant. Recent advancements in overset grids are given by Meakin [85] and others [64, 79]. Wang [147] creates a patch boundary in the overset region in order to achieve conservative flux representation.

2.2 *Unstructured grids*

Unstructured grid generation has its roots in the finite element community of structure modeling. Unstructured grids have inherent simplicity of construction in that, by definition, no structure is required. Also it is not inherently necessary to communicate the actual topology of the configuration to the grid generator. Although largely synonymous with tetrahedral grids, unstructured grids may alternatively be composed of hexahedral cells (without directional structure). The term might strictly encompass any combination of cell shapes, but in the grid generation literature combinations of regions with structure (e.g., structured or prismatic grids near body surfaces) with regions without structure are generally called hybrid grids. For that matter, block-structured grids are unstructured in the large.

There are fundamentally three approaches to the generation of tetrahedral grids that have attracted the most interest [5]: octree [140], Delaunay [4, 6, 80, 88, 97], and advancing front [53, 77, 81, 82]. Other approaches are noted as well in the historical summary by Field [36].

Unstructured grid generation based on the Delaunay triangulation

Most algorithms for generating unstructured-triangular and unstructured-tetrahedral grids are based on the Delaunay triangulation or the so-called "advancing front" method. Unstructured grids consisting of other types of elements than triangles and tetrahedra are not considered here.

The Delaunay triangulation has been used for various applications, including scattered data interpolation. In that context, function values f_i are given at scattered locations $\mathbf{x}_i$ for which no connectivity is given. Typically, the Delaunay triangulation is computed for the points $\mathbf{x}_i$ and triangular (tetrahedral) interpolants are constructed. The Delaunay triangulation is characterized by the fact that the circle (sphere) passing through the vertices of any triangle (tetrahedron) does not contain any point of the original point set in its interior. In the 2D case, the Delaunay triangulation is the max-min angle triangulation of the given point set, i.e., it maximizes the minimum angle in the triangulation [95]. This property is very desirable for many applications, particularly for grid generation.

The Delaunay triangulation has found wide popularity in the finite element method (FEM) and with the unstructured grid generation communities. In general, triangles (tetrahedra) must be generated for a simply connected region in 2D (3D). They are generated in two steps. The first step is the generation of a boundary grid for the boundary curves (boundary surfaces) of the space to be discretized, i.e., a set of boundary conforming line segments (triangles) is computed. The second step is the generation of triangles (tetrahedra) inside this boundary grid [139].

The unstructured grid can be generated by inserting points and performing local retriangulation iteratively or by first generating the entire point set and then triangulating it, possibly using parallel programming paradigms. Grid points are placed such that certain quality measures (e.g., lengths, areas, volumes, angles, ratios thereof, etc.) and specified distributions are satisfied.

The grid points are typically chosen according to geometrical properties of the boundary (e.g., arc length or absolute curvature) and distribution functions. Spacing parameters are used for the boundary point distribution, and so-called "sources" (i.e., point, line, and plane sources) are used to further control grid point distributions in the interior. Grid point densities decrease in a predefined fashion with increasing distance from the sources. Grids must be graded in this fashion due to the sometimes sudden occurrence of discontinuities in certain field parameters, e.g., shocks.

Unstructured grid generation based on element size optimization

An automatic algorithm for the generation of 3D unstructured-tetrahedral grids is described by Hamann et al. [50]. This algorithm can be applied to any closed geometry, i.e., a geometry consisting of surfaces whose boundary curves are each shared by exactly one other surface boundary curve. The method is based on intersecting the edges of an initial (coarse) tetrahedral grid with the given geometry, clipping this initial grid against the geometry by extracting the (parts of) tetrahedra on the outside (or inside) of the geometry, and iteratively inserting grid points.

The initial tetrahedral grid consists of a uniform density of points and tetrahedra, and the triangulation is iteratively improved by inserting grid points until the tetrahedral volumes satisfy a specified condition. This condition considers the distance to and—if desired—the absolute curvature of the geometry. The overall goal is to minimize the difference between actual and desired tetrahedral volumes [50].

Unstructured grid generation based on the advancing front method

The advancing front method generates unstructured grids in a completely different way. The input—for the generation of a triangular (tetrahedral) grid—is a set of oriented line segments (oriented triangles) discretizing the boundary curves (surfaces) of some geometry. Triangles (tetrahedra) are then constructed by advancing a front into the interior of the field until it is completely filled with elements.

Depending on local edge and angle configurations of the current front, new elements are created by connecting existing points or by inserting new points—according to some distribution function—and connecting them with existing ones [78].

The desired point distribution is typically defined by a set of points with associated spacing parameters (background grid). Interpolating these spacing parameters yields the desired spacing at any point in the field. Points are inserted such that the desired spacing is optimally satisfied. The advancing front strategy is also used for the generation of unstructured grids consisting of quadrilaterals (hexahedra).

Unstructured hexahedral grids

Unstructured hexahedral grid generators in 3D are less well-developed, but are progressing. One approach is to base the procedure on triangle/tetrahedral generators, and several works in that direction were noted by Thompson and Weatherill [129]. Another approach that has been pursued for some time is that of paving [13] a 2D region outward from boundaries, with much cutting and stitching. Schneiders and Bunten [102] fill the region with a regular cartesian grid of cubes, stopping a certain minimum distance from the boundaries, and then connect exposed corners of cubes to the boundary. Ives [59] uses the approach of embedding surfaces in a base grid, cutting out intersections, and then snapping the surface into the base grid.

Another fundamental approach to hexahedral grid generation is to simply use the techniques of block-structured grid generation, with the blocks as large unstructured hexahedral "bricks", created by any means available, within which a structured—and therefore inherently hexahedral—grid is generated by TFI or elliptic systems. This approach, with TFI, actually predates later approaches, as noted by Field [36], but its potential may currently be underevaluated.

2.3 Hybrid grids

Both, unstructured tetrahedral grids and recursively subdivided cartesian grids are unacceptably inefficient for high Reynolds number solutions because of the lack of high aspect ratio cells near body surfaces. Although both approaches can do Navier–Stokes solutions, the number of cells required renders the point moot. This same concern arises in any field solution with very large scale variation. The truncation error of a finite volume discretization depends on the shape of the control volume, and difficulties exist also in resolving wakes and other free shear layers. This has naturally led to interest in hybrid grids—tetrahedral or cartesian away from the body surfaces, with some structure such as prisms or hexahedra admitting effective cells with high aspect ratios near those surfaces.

Some recent examples of hybrid grids using prisms are in [23, 64, 66, 70, 81]. Steinbrenner and Noack [122] build a hybrid grid on the Chimera overset grid approach, generating block-structured grids around each body as in Chimera, cutting holes where these grids would overlap, and then filling the holes with tetrahedral cells via advancing front. Kao and Liou [68] and Shaw et al. [104] follow similar approaches.

2.4 Hierarchical grid generation based on quadtrees and octrees

By applying a subdivision technique based on quadtrees (octrees) to certain elements of a grid, multiple levels of different resolution are defined. This approach leads to so-called hierarchical (embedded) grids. Depending on the local complexity of the geometry or the local complexity of an intermediate field solution, more elements are adaptively created by splitting parent elements.

In the structured case, a quadrilateral (hexahedron) is subdivided into four (eight) quadlaterals (hexahedra) which are defined by splitting the edges at their midpoints [83, 101]. The subdivision algorithm must ensure that no "cracks" are introduced in physical space when splitting edges in the parameter space of a parameter space of a parametric surface. As a result of the subdivision process, nodes, usually called "hanging nodes", are created that lie in the interior of edges (faces).

Hierarchical subdivision schemes based on quadtrees (2D) and octrees (3D) are also used for the automatic generation of entire unstructured grids inside simply connected regions. For this purpose, the boundary of the region to be discretized must be given by a set of line segments (triangles), and the bounding box of the region is recursively subdivided until no cuboid contains more than one of the points used in the boundary discretization. The cuboids lying in the interior of the region are then subdivided into triangles (tetrahedra). Special care is necessary to preserve the region's boundary. This approach is discussed by Yerri and Shephard [144–146]. Coirier and Powell [22], Aftosmis et al. [1] and deZeeuw and Powell [32] are recent examples of cartesian grids from this approach.

3 Surface grids

Generation of a satisfactory surface grid remains a major bottleneck in the grid generation process, largely stemming from the fact that CAD systems adequate for numerical machining are not necessarily adequate for numerical simulation. Gatzke et al. [44] at McDonnell Douglas noted that:

> "Experience has shown that one-third to one-half of the time between obtaining point surfaces from CAD and completing the grid is spent manipulating the geometry into the form desired for the application at hand".

Grid generation systems must be able to take CAD data as it comes—with gaps, overlaps, various definitions, etc. The general approach is to cover the incoming CAD patches in some manner with a regular set of patches on which a grid is generated and then transferred to the underlying CAD patches. The use of fundamental concepts from differential geometry to formulate the Laplace–Beltrami generation system for surface grids is discussed by Warsi [138] and applied by Thompson [134]. Some recent innovations are given by Spekreijse et al. [117].

A detailed discussion of elliptic structured grid generation using NURBS surfaces is given by Khamayseh and Hamann [72]. Parametric surfaces/volumes and their relation to grid generation applications are dealt with by Barnhill et al. [7] and others [35, 57]

4 Approximation of geometries with discontinuities

For grid generation purposes, a geometry that is to be discretized must be error-free, i.e., it must not contain discontinuities (overlapping surfaces, "gaps" between surfaces, or surface intersections). Unfortunately, CAD data imported from other CAD systems very often contains such errors.

The National Grid Project (NGP) system provides an interactive geometry correction technique that locally approximates a given geometry by a B-spline surface. The technique must be applied to all regions containing discontinuities. Eventually, a continuous geometry is obtained that consists partially of original NURBS (non-uniform rational B-spline) surfaces and partially of B-spline approximations. The continuous geometry is then used for grid generation.

The geometry correction technique is based on constructing an initial local surface approximation (a bilinearly blended Coons patch) which is projected onto the given surfaces. The user must ensure that the B-spline surface approximations are connected with each other and with certain given NURBS surfaces in a full-face interface fashion. This can be done with the CAD tools provided by the NGP system. Generating a local surface approximation requires these steps:

1. Definition of four surface boundary curves.
2. Generation of $N \times N$ points on the bilinearly blended Coons patch defined by the four surface boundary curves.
3. Projection of the $N \times N$ points onto the given surfaces.

Generation of "artificial projections" whenever certain points of the Coons patch cannot be projected onto any original surface.

When trying to project a point of the Coons patch onto the given surfaces, a projection might or might not be found. If one or more projections is found within a small distance from the Coons patch, the one closest to the Coons patch is chosen. If no projection is found, an "artificial projection" is approximated by applying a scattered data interpolation scheme to the projections that have been found. Hardy's reciprocal multiquadric method is used for interpolating the known values. The resulting $N \times N$ points (projections and "artificial projections") are interpolated by a C^1 continuous, bicubic B-spline surface. An error estimate is computed for each local surface approximant. Curves on the original geometry, e.g., surface boundary curves or trimming curves, can be preserved by this method.

4. Interpolation of the points resulting from steps (2) and (3) by a bicubic B-spline surface.

This algorithm was described in detail [49, 60, 110]. The underlying concepts are well known in geometric modeling [35, 57].

5 Adaptive grids

Grid adaptation is carried out by some combination of redistribution (moving the grid points), refinement (adding/deleting grid points), or modification of the solution representation on the grid. Redistribution has been the favored approach with structured grids, refinement with unstructured grids, and little use has been made of the third approach in real CFD applications. Numerous papers have addressed refinement of unstructured grids, as this is, in fact, a natural strength of that approach. Good examples are found in the Barcelona and Swansea grid conference proceedings and many recent works are noted by Thompson [128]. Other examples are those by Kallinderis and coworkers [65, 67, 91], Aftosmis [2], Davis and Dannenhoffer [27], Marcum and Weatherill [82].

Dynamically adaptive grids used for solutions evolving in time have long been based on physical analogies—springs and such, or continuum mechanics, e.g., [26, 52, 74]. A related approach is movement of grid points toward a weighted center of mass [10].

Variational principle approaches to grid adaptation generally weight measures for smoothness, orthogonality, and clustering in some manner. A number of such elements are collected by Warsi and Thompson [137]. Brackbill [16] extends the formulation to include adaptation to align the grid with a vector field. Knupp [74] also includes alignment with a given vector field. Other recent work is by Hagmeijer [48]. The control functions of elliptic generation also can serve for grid adaptation [73].

Meakin [84] refines a set of cartesian background grids to accommodate moving bodies in the Chimera overset grid approach. Kao et al. [69] use the Chimera approach as an adaptive mechanism, somewhat in the spirit of the AMR (adaptive mesh refinement) with a hierarchy of overset cartesian grids of different levels of refinement. Related approaches are those by Pember et al. [92] and Moore et al. [87].

6 Large grid generation systems

There are now quite a number of large grid generation systems, in fact, nearly every establishment now seems to have at least one of its own. Some of these systems are freely available, some are proprietary, and some are commercial systems. The discussion that follows notes some that are identified in the literature by name. There are others, of course, yet without benefit of such appellation.

The NGP system developed at Mississippi State University [40, 41, 98, 131] is an interactive geometry and grid generation system for block-structured, tetrahedral, and hybrid grids. The system reads CAD data via IGES and converts all surface patches to NURBS format. A "carpet", composed of interfacing NURBS patches, is then placed over the given CAD patches to correct gaps, overlaps, and intersections. The system also has internal CAD capability for the construction or repair of surfaces. Surface grids are generated based on the "carpet", a NURBS aproximation, and can be projected onto the original CAD patches. Both the surface grids and the subsequent volume grids can be generated as block-structured via elliptic, hyperbolic, or TFI methods, or as unstructured grids via the Delaunay or advancing front methods.

The ICEM-CFD system [3, 11, 12, 28, 143] is a commercial system from ICEM-CFD Engineering that was developed in Europe in the late 80s and continues to be enhanced. The system now includes block-structured grids, tetrahedral grids, and unstructured hexahedral grids. The system interfaces with numerous CAD systems and has been connected to a number of flow solvers.

The GRIDGEN system of Pointwise [21, 123–126] is an interactive block-structured system that first emerged from General Dynamics in the late 80s and continues to be RGE enhanced. Current versions are commercial systems from Pointwise. The user constructs curves which are in turn used to build the topological surface and volume components. The user then selects curves as the boundaries of surface grids, and finally surfaces as the boundaries of volume grids (blocks). With this system, grid generation is a user-in-the-loop task. The data structure maintains the relationship among the curves, surfaces, and volumes so that changes can be propagated up or down the hierarchy automatically.

EAGLE View [61, 99, 111] of Mississippi State University incorporates the broad capabilities of EAGLE [42, 132–134] into a graphical interface that allows interactive construction with journaling. The features of this system have been outlined above.

GridPro/az3000 [33, 34], a commercial block-structured program of Program Development Corporation, uses a language—Topology Input Language (TIL)—to define both the surface and the block-structured grid. The language includes components (objects) that can be invoked, and therefore admits the formation of element libraries.

CFD-GEOM of CFD Research Corporation [58] is an interactive geometric modeling and grid generation system for block-structured grids, tetrahedral (advancing front) grids, and hybrid grids. All elements are linked so that updates are propagated throughout the database. The geometry is NURBS-based, reads IGES files, and has some internal CAD capabilities. The system also has macro library capability.

The RAGGS system of Rockwell [142] covers incoming CAD patches with "quilts" of NURBS and/or rational Bezier patches, generates surface grids from edges in space by TFI, and then projects these grids onto the "quilts". The system then generates either structured or unstructured grids.

The MACGS block-structured grid generation system of McDonnell Douglas [43–45, 76] interfaces with CAD systems via IGES, incorporates an internal geometry modification capability, generates surface grids parametrically, and generates volume grids by TFI and elliptic means.

ENGRID is a block-structured system of the National Aerospace Laboratory in the Netherlands, Alenia/Gat, and Fokker [15, 119, 120] incorporating both TFI and elliptic generation.

The CSCMDO system [63] of CSC and NASA Langley is a block-structured volume grid generator developed to allow modifications to geometry by MDO. VOLUME [71] is another NASA Langley system that inputs surfaces from another system and generates a volume grid by TFI.

MBGRID of Canadair is a block-structured grid generation system that operates on a surface block structured formed in a CAD system. The grid generation system includes both TFI and elliptic, and is done with full-face matching among the blocks.

The GEMS block-structured grid generation system SAMTEK-ITC in Turkey [29] is based on object-oriented programming and C++ that uses case-based reasoning and reinforcement learning to capture CDF expertise. The system selects the case that is best suited for a particular geometry from among known ones.

The 3DGRAPE/AL system of NASA Ames is a block-structured grid generator that now includes the specification of arbitrary intersection angles at boundary surfaces, as well as the orthogonality pioneered by earlier versions. 3DPERP [114, 115] is an interactive system that sets up input for 3DGRAPE. GRAPEVINE [113] provides an overall GUI for 3DGRAPE.

The MEGACADS block-structured grid generation system of DLR in Germany [100] also uses the conventional elliptic system with control functions iteratively adjusted to achieve boundary orthogonality with specified off-unstructured spacing.

The UNISG system of Fiat and Rockwell [18, 127] is a block-structured grid generation system that interfaces with CAD systems via IGES files. The system uses both the conventional elliptic system and also a biharmonic system for boundary orthogonality.

The IMESH and ELGEN3 block-structured systems of DLR in Germany [37, 56] use the biharmonic generation system to achieve boundary orthogonality.

The GENIE++ block-structured grid generation system of Mississippi State University [111] was also introduced in the late 80s and has been continually enhanced over the years. This system uses TFI with elliptic smoothing and includes various splining methods.

The INGRID block-structured system of Deutsche Airbus [8] cuts local blocks out of a global mesh and fills the space with new component grids.

The TIGER system of Mississippi State University [106, 107, 111, 112] is a block-structured system specialized for turbomachinery applications. The system uses TFI with elliptic smoothing and NURBS boundary representation.

The RAPID system of NASA Langley [109] is specialized to a class of airplane configurations. The CAMP system of Lockheed [103] is a block-structured system specialized to aircraft configurations in a modular approach.

The GRID* block-structured system of the European Space Agency [54] follows the Unix toolbox concept and consists of a collection of C routines, fairly small in size. Connectivity and orientations are specified by an input file.

The AGPS surface geometry of Boeing [17, 108] is a surface preparation program than can create the input deck for EAGLE. RAMBO-4G of Aerospace [136] also can set up an EAGLE input deck.

The SAUNA hybrid grid generation system of the Aircraft Research Association in the U.K. [105] allows the user to remove sections of the block-structured grid, with triangular grids being then generated on the surfaces so exposed, followed by filling of the void with tetrahedral grids.

The IGG system of Virje Universiteit in Belgium [30, 31] is a block-structured system that uses TFI and elliptic generation, as well as advancing front to generate hybrid grids.

VGRID of NASA Langley and VIGYAN [39, 90, 94] is a tetrahedral grid generator which uses advancing front with a cartesian background grid to control resolution.

TGrid of Fluent [14] is a tetrahedral grid generator based on the Delaunay approach.

The first general purpose domain connectivity programs for Chimera grids were the PEGSUS (from the Air Force AEDC) and CMPGRD (from IBM) programs in the late 80s [85], which continue to be enhanced. Advances in CMPGRD are detailed by Henshaw et al. [55]. Later programs are DCF3D of NASA Ames and Overset Methods [86] and BEGGAR of the Air Force Wright Laboratory at Eglin [9, 79]. The FAME system of the Defence Research Agency in the U.K. [89] is another.

7 Conclusion

There are still a number of approaches being pursued in grid generation, with some combinations also. Block-structured grids require less storage per grid point and can take advantage of factored and directional solvers. Tetrahedral grids work well for Euler solutions but not for Navier–Stokes solvers. Chimera grids are versatile and have some definite advantages for bodies in relative motion, but concerns arise at interfaces. Hybrid grids combine the best features of structured and unstructured grids. And cartesian grids continue to hang in there in one form or another.

The major driving factors in comprehensive grid systems must first be automation and then graphical interaction. Since design is the paramount application, the efficiency of a grid generation system is measured primarily by the person-time it takes to generate a series of geometrically related grids for complex configurations. And the coupling with CAD systems on the front end and with solution and visualization systems on the back end must be smooth and effective. The ideal is not to make it easy for a person to generate a grid but rather to remove the person from the process—not to make it interactive, but to make it automatic.

Grid generation tools must be designed to be applied by design engineers rather than grid generation specialists.

Grid generation systems must be capable of handling very large scale variations, as occur in high Reynolds number flows, and this precludes any approach not encompassing large aspect ratio cells with good numerical properties.

Finally, there is a clear need for interaction with commercial CAD vendors. CAD systems were developed before the advance of grid generation technology and widespread application. In order to become truly effective in multidisciplinary design optimization, CAD tools must be redesigned to target computational analysis as well as numerical machining and material formation.

References

1. Aftosmis, M.J., Melton, J.E., Berger, M.J.: Adaptation and surface modeling for cartesian mesh methods. In: 12th AIAA computational Fluid Dynamics Conference, San Diego, CA, June 1995, AIAA-95-1725-CP (1995).

2. Aftosmis, M.J.: Upwind method for simulation of viscous flow on adaptively refined meshes. AIAA J. 32: 268–277 (1994).
3. Akdag, V., Wulf, A.: Integrated geometry and grid generation system for complex configurations. Smith, R.E. (ed.): Proceedings of the Software Systems for Surface Modeling and Grid Generation Workshop. NASA Langley Research Center, Hampton, VA, p. 161 (NASA conference publication 3143) (1992).
4. Anderson, W.K.: A grid generation and flow solution method for the Euler equations in unstructured grids. J. Comput. Phy. 110: 23–38 (1994).
5. Baker, T.J.: Prospects and expectations for unstructured methods. In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 273 (NASA conference publication 3291) (1995).
6. Baker, T.J.: Point placement and control of triangle quality for inviscid and viscous mesh generation. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Conference. Pineridge Press, Swansea, Wales, pp. 137–149 (1994).
7. Barnhill, R.E., Farin, G., Hamann, B.: NURBS and grid generation. In: Babuska, I., Flaherty, J.E., Henshaw, W.D., Hopcroft, J.E., Oliger, J.E., Tezduyar, T. (eds.): Modeling, mesh generation, and adaptive numerical methods for partial differential equations. Springer, Berlin Heidelberg New York Tokyo pp. 1 (IMA volumes in mathematics and its applications, vol. 75) (1995).
8. Becker, K.: Interactive algebraic mesh generation for Twin jet transport aircraft. Arcilla, A.S., Hauser, J., Eiseman, P.R., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 3rd Internationl Grid Conference. North-Holland, Amsterdam, pp. 455–466 (1991).
9. Belk, D.M.: The role of overset grids in the development of the general purpose CFD code. In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 193–204 (NASA conference publication 3291) (1995).
10. Benson, R.A., McRae, D.S.: Time-accurate simulations of unsteady flows with a dynamic solution-adaptive grid algorithm. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Conference. Pineridge Press, Swansea, Wales, pp. 573–588 (1994).
11. Bertin, D., Casties, C., Lordon, J.: A new automatic grid generation environment for CFD applications. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Conference. Pineridge Press, Swansea, Wales, pp. 391–402 (1994).
12. Bertin, D., Lordon, J., Moreux, V.: A new automatic grid generation environment for CFD applications. In: 10th AIAA Applied Aerodynamics Conference, Palo Alto, CA, June 1992, IAA-92-2720-CP (1992).
13. Blacker, T.D., Stephenson, M.B.: Paving: a new approach to automated quadrilateral mesh generation. Int. J. Numer. Methods Eng. 32: 811 (1991).
14. Blake, K.R., Spragle, G.S.: Unstructured 3D Delaunay mesh generation applied to planes, trains and automobiles. In: 31st AIAA Aerospace Sciences Meeting, Reno, NV, January 1993, AIAA-93-0673, (1993).
15. Boerstoel, J.W., Spekreijse, S.P.: An information system for the numerical simulation of 3D Euler flows around aircraft. Comput. Methods Appl. Mech. Eng. 89: 237–244 (1991).
16. Brackbill, J.U.: An adaptive grid with directional control. J. Comput. Phys. 108: 38–50 (1993).
17. Capron, W.K., Smit, K.L.: Advanced aerodynamic applications of an interactive geometry and visualization system. In: 29th AIAA Aerospace Sciences Meeting, Reno, NV, January 1991, AIAA-91-0800 (1991).
18. Casella, M., Vitali, D.F., Bergamini, P., Szema, K.Y., Ramakrishan, S.V.: A multi-block structured grid interactive package for automotive industry CFD applications. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Generation Conference. Pineridge Press, Swansea, Wales, pp. 403–415 (1994).
19. Castillo, J.E.: Mathematical aspects of numerical grid generation. SIAM, Philadelphia (1991).
20. Chan, W.M., Steger, J.L.: Enhancements of a three-dimensional hyperbolic grid generation scheme. Appl. Math. Comput. 51: 181–205 (1992).
21. Chawner, J.R., Steinbrenner, J.P.: Automatic structured grid generation using GRIDGEN (some restrictions apply). In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 463–476 (NASA conference publication 3291) (1995).

22. Coirier, W.J., Powell, K.G.: A cartesian, cell-based approach for adaptively-refined solutions of the Euler and Navier–Stokes equations. In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 207–224 (NASA conference publication 3291) (1995).
23. Connell, S.D., Braaten, M.E.: Semistructured mesh generation for three-dimensional Navier–Stokes calculations. AIAA J. 33: 1017–1024 (1995).
24. Connell, S.D., Sober, J.S., Lamson, S.H.: Grid generation and surface modeling for CFD. In: Proceedings of the Surface Modeling, Grid Generation, and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 29–43 (NASA conference publication 3291) (1995).
25. Cosner, R.R.: Future requirements in surface modeling and grid generation. In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 3–13 (NASA conference publication 3291) (1995).
26. Davies, C.B., Venkatapathy, E.: Application of solution adaptive grid scheme to complex three-dimensional flows. AIAA J. 30: 2227–2233 (1992).
27. Davis, R.L., Dannenhoffer, J.F.: Three-dimensional adaptive grid-embedding Euler technique. AIAA J. 32: 1167–1174 (1994).
28. Dela Viuda, J.M., Diet, J., Ranoux, G.: Path-independent structured multiblock grids for CFD computations, In: Arcilla, A.S., Hauser, J., Eiseman, P.R., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 3rd International Grid Conference. North-Holland, Amsterdam, pp. 703–715 (1991).
29. Dener, C., Koc, E., Sirin, I.: Extension to GEMS for automatic grid generation and intelligent topology definition. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation on computational field simulation and related fields: Proceedings of the 4th International Grid Conference. Pineridge Press, Swansea, Wales, pp. 453–464 (1994).
30. Dener, C., Hirsch, C.: IGG: an interactive 3D surface modelling and grid generation system. In: 30th AIAA Aerospace Sciences Meeting, Reno, NV, January 1992, AIAA-92-0073 (1992).
31. Dener, C., Hirsch, C.: IGG: an advanced interactive grid generation system. In: Arcilla, A.S., Hauser, J., Eiseman, P.R., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 3rd International Grid Conference. North-Holland, Amsterdam, pp. 503–518 (1991).
32. deZeeuw, D., Powell, K.G.: An adaptively refined cartesian mesh solver for the Euler equations. J. Comput. Phys. 104: 56–68 (1993).
33. Eiseman, P.R.: Multiblock grid generation with automatic zoning In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 143–162 (NASA conference publication 3291) (1995).
34. Eiseman, P.R., Cheng, Z., Hauser, J.: Applications of multiblock grids generations with automatic zoning. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Conference. Pineridge Press, Swansea, Wales, pp. 123–134 (1994).
35. Farin, G.: Curves and surfaces for computer aided geometric design, 3rd edn. Academic Press, San Diego (1993).
36. Field, D.A.: The legacy of automatic mesh generation from solid modeling. Comput. Aided Geom. Des. 12: 651–673 (1995).
37. Findling, A., Herrmann, U.: Development of an efficient and robust solver for elliptic grid generation. In: Arcilla, A.S., Hauser, J., Eiseman, P.R., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 3rd International Grid Generation Conference. North-Holland, Amsterdam, pp. 781–792 (1991).
38. Franke, R.: Scattered data interpolation: tests of some methods. Math. Comput. 38: 181–200 (1982).
39. Frink, N.T., Pirzadeh, S., Parikh, P.: An unstructured-grid software system for solving complex aerodynamic problems. In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 289–308 (NASA conference publication 3291) (1995).
40. Gaither, A., Gaither, K., Jean, B., Remotigue, J., Whitmire, J., Soni, B.K., Thompson, J.F., Danenhoffer, J.F., Weatherill, N.P.: The National Grid Project: a system overview. In: Proceedings of the surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 423–446 (NASA conference publication 3291) (1995).

41. Gaither, A.: A topology model for numerical grid generation. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Conference. Pineridge Press, Swansea, Wales, pp. 247–258 (1994).
42. Gatlin, B., Thompson, J.F., Yoon, Y.-H., Luong, P.V., Ganapathiraju, D., Wolverton, M.K.: Extensions to the EAGLE grid code for quality control and efficiency. In: 29th AIAA Aerospace Sciences Meeting, Reno, NV, January 1991, AIAA-90-0148 (1991).
43. Gatzke, T.D., Melson, T.G.: Generating grids directly on CAD database surfaces using a parametric evaluator approach. In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 505–515 (NASA conference publication 3291) (1995).
44. Gatzke, T.D., LaBozzetta, W.F., Cooley, J.W., Finfrock, G.P.: Geometry acquisition and grid generation—recent experiences with complex aircraft configurations. In: Smith, R.E. (ed.): Proceedings of the Software Systems for Surface Modeling and Grid Generation Workshop. NASA Langley Research Center, Hampton, VA, pp. 31–43 (NASA conference publication 3143) (1992).
45. Gatzke, T.D., LaBozzetta, W.F., Finfrock, G.P., Johnson, J.A., Romer, W.W.: MACGS: a zonal grid generation system for complex aero propulsion configurations. AIAA-91-2156 (1991).
46. George, P.L.: Automatic mesh generation. Wiley, New York (1991).
47. Gordon, W.J.: Blending-function methods of bivariate and multivariate interpolation and approximation. SIAM J. Numer. Anal. 8: 158–177 (1971).
48. Hagmeijer, R.: Grid adaption based on modified anisotropic diffusion equations formulated in the parametric domain. J. Comput. Phys. 115: 169–183 (1994).
49. Hamann, B.: Construction of B-spline approximations for use in numerical grid generation. Appl. Math. Comput. 65: 295–314 (1994).
50. Hamann, B., Chen, J.L., Hong, G.: Automatic generation of unstructured grids for volumes outside or inside closed surfaces. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Generation Conference. Pineridge Press, Swansea, Wales, pp. 187–197 (1994).
51. Hamann, B., Sarraga, R.F. (eds.): Finite elements, grid generation and geometric design. Comput. Aided Geom. Des. 12, special issue (1995).
52. Harvey, A.D., Acharya, S., Lawrence, S.L.: Space marching calculations about hypersonic configurations using a solution-adaptive mesh algorithm. AIAA J. 31: 1809 (1993).
53. Hassan, O., Probert, E.J., Morgan, K., Peraire, J.: Unstructured mesh generation for viscous high speed flows. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Conference. Pineridge Press, Swansea, Wales, pp. 779–793 (1994).
54. Hauser, J., Paap, H.G., Wong, H., Spel, M.: GRID*: a general multiblock surface and volume grid generation toolbox. In: Arcilla, A.S., Hauser, J., Eiseman, P.R., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 3rd International Grid Conference. North-Holland, Amsterdam, pp. 817–837 (1991).
55. Henshaw, W.D., Chessire, G., Henderson, M.E.: On constructing three-dimensional overlapping grids with CMPGRD. In: Smith, R.E. (ed.): Proceedings of the Software Systems for Surface Modeling and Grid Generation Workshop. NASA Langley Research Center, Hampton, VA, pp. 415–434 (NASA conference publication 3143) (1992).
56. Herrmann, U.: IMESH: an interactive mesh generation package for graphics super workstations. In: Arcilla, A.S., Hauser, J., Eiseman, P.R., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 3rd International Grid Conference. North-Holland, Amsterdam, pp. 467–478 (1991).
57. Hoscheck, J., Lasser, D.: Fundamentals of computer aided geometric design. A.K. Peters, Wellesley, MA (1993).
58. Hufford, G.S., Harrand, V.J., Patel, B.C., Mitchell, C.R.: Evaluation of grid generation technologies from an applied perspective. In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 401–420 (NASA conference publication 3291) (1995).
59. Ives, D.: Geometric grid generation. In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 535–557 (NASA conference publication 3291) (1995).
60. Jean, B.A., Hamann, B.: Interactive techniques for correcting CAD/CAM data. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Generation Conference. Pineridge Press, Swansea, Wales, pp. 317–328 (1994).

61. Jiang, M.-Y., Remotigue, M.G., Stokes, M.L., Thompson, J.F.: EAGLEView: grid enhancement and applications. In: 32nd AIAA Aerospace Sciences Meeting, Reno, NV, January 1994, AIAA-94-0316 (1994).
62. Johnson, R.A., Belk, D.M.: Multigrid approach to overset grid communication. AIAA J. 33: 2305–2308 (1995).
63. Jones, W.T., Samareh–Abolhassani, J.: A grid generation system for multi-disciplinary design optimization. In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 657–667 (NASA conference publication 3291) (1995).
64. Kallinderis, Y., Khawaja, A., McMorris, H.: Hybrid prismatic/tetrahedral grid generation for complex geometries. In: 33rd AIAA Aerospace Sciences Meeting, Reno, NV, January 1995, AIAA-95-0211 (1995).
65. Kallinderis, Y., Nakajima, K.: Finite element method for incompressible viscous flows with adaptive hybrid grids. AIAA J. 32: 1617–1625 (1994).
66. Kallinderis, Y., Ward, S.: Prismatic grid generation for three-dimensional complex geometries. AIAA J. 31: 1850–1856 (1993).
67. Kallinderis, Y., Vijayan, P.: Adaptive refinement–coarsening scheme for three-dimensional unstructured meshes. AIAA J. 31: 1440–1447 (1993).
68. Kao, K.-H., Liou, M.-S.: Advance in overset grid schemes: from Chimera to DRAGON grids. AIAA J. 33: 1809–1815 (1995).
69. Kao, K.-H., Liou, M.-S., Chow, C.-Y.: Grid adaptation using chimera composite overlapping meshes. AIAA J. 32: 942–949 (1994).
70. Karman, S.L.: Unstructured cartesian/prismatic grid generation for complex geometries. In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 251–270 (NASA conference publication 3291) (1995).
71. Kerr, P.A., Smith, R.E., Posenau, M.A.: GEOmetry LABoratory (GEOLAB) surface modeling and grid generation technology and services. In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 69–80 (NASA conference publication 3291) (1995).
72. Khamayseh, A., Hamann, B.: Elliptic grid generation using NURBS surfaces. Comput. Aided Geom. Des. 13: 369–386 (1996).
73. Kim, H.J., Thompson, J.F.: Three dimensional adaptive grid generation on a composite block grid. AIAA J. 28: 470–477 (1990).
74. Knupp, P.: Mesh generation using vector-fields. J. Comput. Phys. 119: 142–148 (1995).
75. Knupp, P., Steinberg, S.: Fundamentals of grid generation. CRC Press, Boca Raton (1993).
76. LaBozzetta, W.F., Gatzke, T.D., Ellison, S., Finfrock, G.P., Fisher, M.S.: MACGS: towards the complete grid generation system. In: 12th AIAA Applied Aerodynamics Conference, Colorado, Springs, CO, June 1994, AIAA-94-1923 (1994).
77. Lohner, R.: Matching semi-structured and unstructured grids for Navier–Stokes calculations. In: 11th AIAA Computational Fluid Dynamics Conference, Orlando, FL, July 1993, AIAA-93-3348-CP (1993).
78. Lohner, R., Parikh, P.: Three-dimensional grid generation by the advancing front method . Int. J. Numer. Methods Fluids 8: 1135–1149 (1988).
79. Maple, R.C., Belk, D.M.: Automated setup of blocked, patched and embedded grids in the beggar flow solver. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Conference. Pineridge Press, Swansea, Wales, pp. 305–314 (1994).
80. Marchant, M.J., Weatherill, N.P.: Unstructured grid generation for viscous flow simulations. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Conference. Pineridge Press, Swansea, Wales, pp. 151–162 (1994).
81. Marcum, D.L.: Generation of unstructured grids for viscous flow applications. In: 33rd AIAA Aerospace Sciences Meeting, Reno, NV, January 1995, AIAA-95-1722 (1995).
82. Marcum, D.L., Weatherill, N.P.: Aerospace applications of solution adaptive finite element analysis. Comput. Aided Geom. Des. 12: 709–731, (1995).
83. Meagher, D.: Geometric modelling using octree encoding. Comput. Graphics Image Process. 19: 129–147 (1982).
84. Meakin, R.L.: An efficient means of adaptive refinement within systems off verset grids. In: 12th AIAA Computational Fluid Dynamics Conference, San Diego, CA, June 1995, AIAA-95-1722 (1995).

85. Meakin, R.L.: Grid related issues for state and dynamic geometry problems using systems of overset structured grids. In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics. NASA Lewis Research Center, Cleveland, OH, pp. 181–192 (NASA conference publication 3291) (1995).
86. Meakin, R.L.: A new method for establishing intergrid communication among systems of overset grids. In: 10th AIAA Computational Fluid Dynamics Conference, Honolulu, HI, June 1991, AIAA-91-1586-CP (1991).
87. Moore, P.K., Flaherty, J.E.: Adaptive local overlapping grid methods for parabolic systems in two space dimensions. J. Comput. Phys. 98: 54–63 (1992).
88. Muller, J.-D.: Quality estimates and stretched meshes based on Delaunay triangulations. AIAA J. 32: 2372–2379 (1994).
89. Onslow, S.H., Blaylock, T.A., Albone, C.M., Hodges, J.: The FAME mesh generation system—its synthesis, flexibility and capabilities. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Conference. Pineridge Press, Swansea, Wales, pp. 259–269 (1994).
90. Parikh, P., Pirzadeh, S.: Recent advances in unstructured grid generation. In: Smith, R.E. (ed.): Proceedings of the Software Systems for Surface Modeling and Grid Generation Workshop. NASA Langley Research Center, Hampton, VA, pp. 435–448 (NASA conference publication 3143) (1992).
91. Parthasarathy, V.N., Kallinderis, Y.: Directional viscous multigrid using adaptive prismatic meshes. AIAA J. 33: 69–78 (1995).
92. Pember, R.B., Bell, J.B., Colella, P.S., Crutchfield, W.Y., Welcome, M.L.: An adaptive cartesian grid method for unsteady compressible flow in irregular regions. J. Comput. Phy. 120: 278–304 (1995).
93. Piperni, P.: Multi-block grid generation with CAD-based domain decomposition techniques. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Conference. Pineridge Press, Swansea, Wales, pp. 109–121 (1994).
94. Pirzadeh, S.: Recent progress in unstructured grid generation. In: 30th AIAA Aerospace Sciences Meeting, Reno, NV, January 1992, AIAA-92-0445 (1992).
95. Preparata, F.P., Shamos, M.I.: Computational geometry, 3rd printing. Springer, Berlin Heidelberg Tokyo (Texts and monographs in computer science) (1990).
96. Raj, P.: Requirements for effective use of CFD in aerospace design. In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 15–28 (NASA conference publication 3291) (1995).
97. Rebay, S.: Effecient unstructured mesh generation by means of Delaunay triangulation and Bowyer–Watson algorithm. J. Comput. Phys. 106: 125–138 (1993).
98. Remotigue, M.G.: The National Grid Project: making dreams into reality. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Conference. Pineridge Press, Swansea, Wales, pp. 429–439 (1994).
99. Remotigue, M.G., Hart, E.T., Stokes, M.L.: EAGLEView: a surface and grid generation program and its data management. In: Smith, R.E. (ed.): Proceedings of the Software Systems for Surface Modeling and Grid Generation Workshop. NASA Langley Research Center, Hampton, VA, pp. 243–252 (NASA conference publication 3291) (1992).
100. Ronzheimer, A., Brodersen, O., Rudnik, R., Findling, A., Rossow, C.-C.: A new interactive tool for the management of grid generation processes around arbitrary configurations. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Conference. Pineridge Press, Swansea, Wales, pp. 441–452 (1994).
101. Samet, H.: The quadtree and related hierarchical data structures. Comput. Surv. 16: 187–285 (1984).
102. Schneiders, R., Bunten, R.: Automatic generation of hexahedral finite element meshes. Comput. Aided Geom. Des. 12: 693–707 (1995).
103. Schuster, D.M.: Batch mode grid generation: an endangered species? In: Smith, R.E. (ed.): Proceedings of the Software Systems for Surface Modeling and Grid Generation Workshop. NASA Langley Research Center, Hampton, VA, pp. 487–500 (NASA conference publication 3143) (1992).

104. Shaw, J.A., Georgala, J.M., Childs, P.N.: General procedures employed in the generation of three-dimensional hybrid structured/unstructured meshes. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Conference. Pineridge Press, Swansea, Wales, pp. 297–303 (1994).
105. Shaw, J.A., Georgala, J.M., May, N.E., Pocock, M.F.: Application of three-dimensional hybrid structured/unstructured grids to land, sea and air vehicles. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Conference. Pineridge Press, Swansea, Wales, pp. 687–698 (1994)
106. Shih, M.-H., Soni, B.K., Yu, T.-Y., Shaunak, S.: Turbomachinery simulation system. In: Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Conference. Pineridge, Press, Swansea, Wales, pp. 417–428 (1994).
107. Shih, M.-H., Soni, B.K.: Geometry modeling and multi-block grid generation for turbomachinery configurations. In: Smith, R.E. (ed.): Proceedings of the Software Systems for Surface Modeling and Grid Generation Workshop. NASA Langley Research Center, Hampton, VA, pp. 463–475 NASA conference publication 3143 (1992).
108. Smit, K.L., Su, T.Y.: A geometry-integrated approach to multiblock grid generation. In: Arcilla, A.S., Hauser, J., Eiseman, P.R., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 3rd International Grid Conference. North-Holland, Amsterdam, pp. 795–803 (1991).
109. Smith, R.J., Bloor, M.I.G., Wilson, M.J., Thomas, A.M.: Rapid airplane parametric input design (RAPID). In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 15–28 (NASA conference publication 3191) (1995).
110. Soni, B., Hamann, B.: Computational geometry tools in grid generation. In: Wang, S.S.Y. (ed.): Advances in hydro-science and engineering, vol. 1, part B. University of Mississippi Press, University, MS, pp. 2004–2009 (1993).
111. Soni, B.K., Thompson, J.F., Stokes, M.L., Shih, M.-H.: GENIE^{++}, EAGLEView and TIGER: general and special purpose graphically interactive grid systems. In: 30th AIAA Aerospace Sciences Meeting, Reno, NV, January 1992, AIAA-92-0071 (1992).
112. Soni, B.K., Shih, M.-H.: TIGER: turbomachinery interactive grid generation. In: Arcilla, A.S., Hauser, J., Eiseman, P.R., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 3rd International Grid Conference. North-Holland, Amsterdam, pp. 315–328 (1991).
113. Sorenson, R.L., McCann, K.: GRAPEVINE: Grids about anything by Poisson's equation in a visually interactive networking environment. In: Smith, R.E. (ed.): Proceedings of the Software Systems for Surface Modeling and Grid Generation Workshop. NASA Langley Research Center, Hampton, VA, pp. 319–331 (NASA conference publication 3143) (1992).
114. Sorenson, R.L., McCann, K.M.: A method for interactive specification of multiple-block topologies. In: Arcilla, A.S., Hauser, J., Eiseman, P.R., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 3rd International Grid Generation Conference. North-Holland, Amsterdam, pp. 731–742 (1991).
115. Sorenson, R.L., McCann, K.M.: A method for interactive specifications of multiple-block topologies. In: 29th AIAA Aerospace Sciences Meeting, Reno, NV, January 1991, AIAA-91-0147 (1991).
116. Sparis, P.D., Karkanis, A.: Boundary-orthogonal biharmonic grids via preconditioned gradient methods. AIAA J. 30: 671–678 (1992).
117. Spekreijse, S.P., Nijhuis, G.H., Boerstoel, J.W.: Elliptic surface grid generation on minimal and parametrized surfaces. In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 617–636 (NASA conference publication 3291) (1995).
118. Spekreijse, S.P.: Elliptic grid generation based on Laplace equations and algebraic transformation. J. Comput. Phys. 118: 38–61 (1995).
119. Spekreijse, S.P., Boerstoel, J.W., Vitagliano, P.L., Kuyvenhoven, J.L.: Domain modeling and grid generation for multi-block structured grids with application to aerodynamic and hydrodynamic configurations. In: Smith, R.E. (ed.): Proceedings of the Software Systems for Surface Modeling and Grid Generation Workshop. NASA Langley Research Center, Hampton, VA, pp. 207–229 (NASA conference publication 3143) (1992).

120. Spekreijse, S.P., Boerstoel, J.W., Vitagliano, P.L.: New concepts for multi-block grid generation for flow domains around complex aerodynamic configurations. In: Arcilla, A.S., Hauser, J., Eiseman, P.R., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 3rd International Grid Conference. North-Holland, Amsterdam, pp. 719–730 (1991).
121. Steger, J.L.: Grid generation with hyperbolic partial differential equations for application to complex configurations. In: Arcilla, A.S., Hauser, J., Eiseman, P.R., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 4th International Grid Conference. North-Holland, Amsterdam, pp. 871–886 (1991).
122. Steinbrenner, J.P., Noack, R.W.: Three-dimensional hybrid grid generation using advancing front techniques. In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 333–356 (NASA conference publication 3291) (1995).
123. Steinbrenner, J.P., Chawner, J.R.: Incorporation of a hierarchical grid component structure into GRIDGEN. In: 31st AIAA Aerospace Sciences Meeting, Reno, NV, January 1993, AIAA-93-0429 (1993).
124. Steinbrenner, J.P., Chawner, J.R.: Recent enhancements to the GRIDGEN structured grid generation system. In: Smith, R.E. (ed.): Proceedings of the Software Systems for Surface Modeling and Grid Generation Workshop. NASA Langley Research Center, Hampton, VA, pp. 253–271 (NASA conference publication 3143) (1992).
125. Steinbrenner, J.P., Chawner, J.R., Anderson, D.A.: Enhancements to the GRIDGEN system for increased user efficiency and grid quality. In: 30th AIAA Aerospace Sciences Meeting, Reno, NV, January 1992, AIAA-92-0662 (1992).
126. Steinbrenner, J.P., Chawner, J.R., Fouts, C.L.: Multiple block grid generation in the interactive environment. In: 21st AIAA Fluid Dynamics, Plasma Dynamics and Lasers Conference, Seattle, WA, 1990, AIAA-90-1602 (1990).
127. Szema, K.-Y.: UNISG: An interactive multizone structured grid generator. In: 11th AIAA Applied Aerodynamics Conference, Monterey, CA, August 1993, AIAA-93-3525 (1993).
128. Thompson, J.F.: A reflection on grid generation in the 90s: trends, needs, and influences. In: Soni, B.K., Hauser, J., Eiseman, P.R., Thompson, J.F. (eds.): Numerical grid generation in computational field simulation and related fields: Proceedings of the 5th International Grid Conference. North-Holland, Amsterdam, pp. 1029–1110 (1996).
129. Thompson, J.E., Weatherill, N.P.: Aspects of numerical grid generation: current science and art. 11th AIAA Aerodynamics Conference, Monterey, CA, August 1993, AIAA-93-3539-CP (1993).
130. Thompson, J.F., Weatherill, N.P.: Structured and unstructured grid generation. In: Pilkington, T.C., Loftis, B., Thompson, J.F. (eds.): High performance computing in biomedical research. CRC Press, Boca Raton, pp. 63–111 (1992).
131. Thompson, J.F.: The National Grid Project. Comput. Syst. Eng. 3: 393–399 (1992).
132. Thompson, J.F., Lijewski, L.E., Gatlin, B.: Efficient application techniques of the EAGLE Grid code to complex missile configurations. In: 27th AIAA Aerospace Sciences Meeting, Reno, NV, January 1989, AIAA-89-0361 (1989).
133. Thompson, J.F.: A composite grid generation code for general 3-D regions. In: 25th AIAA Aerospace Sciences Meetings, Reno, NV, January 1987, AIAA-87-0275 (1987).
134. Thompson, J.F.: A general three-dimensional elliptic grid generation system on a composite block structure. Comput. Methods Appl. Mech. Eng. 64: 377–384 (1987).
135. Thompson, J.F., Warsi, Z.U.A., Mastin, C.W.: Numerical grid generation: foundations and applications. North-Holland, Amsterdam (1985).
136. Visich, M.: Advanced interactive grid generation using RAMBO-4G. In: 29th AIAA Aerospace Sciences Meeting, Reno, January 1991, AIAA-91-0799 (1991).
137. Warsi, Z.U.A., Thompson, J.F.: Application of variational methods in the fixed and adaptive grid generation. Comput. Math. Appl. 19: 31 (1990).
138. Warsi, Z.U.A.: Theoretical foundation of the equations for the generation of surface coordinates. AIAA J. 28: 1140–1142 (1990).
139. Weatherill, N.P., Hassan, O., Marcum, D.L., Marchant, M.J.: Grid generation by the Delaunay triangulation. Von Karman institute for fluid dynamics lecture series 1993–1994, Von Karman Institute for Fluid Dynamics, Rhode-St-Genese, Belgium.
140. Webster, B.E., Shephard, M.S., Rusak, Z., Flaherty, J.E.: Automated adaptive time-discontinuous finite element method for unsteady compressible airfoil aerodynamics. AIAA J. 32: 748–757 (1994).

141. Woan, C.-J., Clever, W.C., Tam, C.K.: Grid generation on trimmed Bezier and NURBS quilted surfaces. In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 517–534 (NASA conference publication 3291) (1995).
142. Woan, C.-J., Clever, W.C., Tam, C.K.: RAGGS: Rockwell automated grid generation system. In: 32nd AIAA Aerospace Sciences Meeting, Reno, NV, January 1994, AIAA-94-0206 (1994).
143. Wulf, A., Akdag, V.: Tuned grid generation with ICEM CFD. In: Proceedings of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop. NASA Lewis Research Center, Cleveland, OH, pp. 477–488 (NASA conference publication 3291) (1995).
144. Yerri, M.A., Shephard, M.S.: A modified quadtree approach to finite element mesh generation. IEEE Comput. Graphics Appl. 3: 39–46 (1983).
145. Yerri, M.A., Shephard, M.S.: Automatic 3D mesh generation by the modified–octree technique. Int. J. Numer. Methods Eng. 20: 1965–1990 (1984).
146. Yerri, M.A., Shephard, M.S.: Automatic mesh generation for three-dimensional solids. Comput. Struct. 20: 31–39 (1985).
147. Wang. Z.J.: A fully conservative interface algorithm for overlapped grids. J. Comput. Phys. 122: 96–106 (1995).

Authors' addresses: J.F. Thompson, NSF Engineering Research Center for Computational Field Simulation, Mississippi State University, P.O. Box 9627, Mississippi State, MS 39762, U.S.A. — B. Hamann, Department of Computer Science, University of California, Davis, California, U.S.A.

Communicated by H. Hagen